Extending Weighting Models with a Term Quality Measure

Christina Lioma and Iadh Ounis

University of Glasgow, Scotland G12 8QQ, U.K.

Abstract. Weighting models use lexical statistics, such as term frequencies, to derive term weights, which are used to estimate the relevance of a document to a query. Apart from the removal of stopwords, there is no other consideration of the quality of words that are being 'weighted'. It is often assumed that term frequency is a good indicator for a decision to be made as to how relevant a document is to a query. Our intuition is that raw term frequency could be enhanced to better discriminate between terms. To do so, we propose using non-lexical features to predict the 'quality' of words, before they are weighted for retrieval. Specifically, we show how parts of speech (e.g. nouns, verbs) can help estimate how informative a word generally is, regardless of its relevance to a query/document. Experimental results with two standard TREC[1] collections show that integrating the proposed term quality to two established weighting models enhances retrieval performance, over a baseline that uses the original weighting models, at all times.

1 Introduction

The aim of an Information Retrieval (IR) system is to retrieve relevant documents in response to a user need, which is usually expressed as a query. The retrieved documents are returned to the user in decreasing order of relevance, which is typically determined by weighting models. Most weighting models use term statistics, such as term frequency, to assign weights to individual terms, which represent the contribution of the term to the document content. These term weights are then used to estimate the similarity between queries and documents [18].

The underlying idea of most weighting models is to boost the weight of terms that occur frequently in a document and rarely in the rest of the collection of documents [18]. Various extensions have been applied on top of this, such as normalising term frequency according to document length [1,5,14], or using the term frequency in specific fields of structured documents (e.g. title, abstract) [8,13,15]. Further extensions include integrating query-independent evidence (e.g. PageRank [2]) to the weighting model in the form of prior probabilities [4,6,12] ('prior' because they are known before the query is issued). For example, assuming that a document's PageRank indicates its quality, integrating PageRank priors to the

[1] Text REtrieval Conference: http://trec.nist.org/

N. Ziviani and R. Baeza-Yates (Eds.): SPIRE 2007, LNCS 4726, pp. 205–216, 2007.

weighting model consists in using information about the document quality when computing how relevant that document is to a query.

We propose a measure of term quality, which is similar to the notion of document quality, in that it is known prior to a query. In addition, our proposed term quality is known prior to the document as well, because it represents how informative a term is generally in language, not with respect to a query or its contribution to the content of a specific document. Hence, this is an intrinsic notion of term quality. The intuition behind using it in IR is that integrating it into the term weighting process may enhance retrieval performance, similarly to the way document quality (in the form of priors, for example) can improve performance when integrated to document ranking [4,6,12].

We derive the proposed term quality in an empirical way from part of speech (POS) n-grams. POS n-grams are n-grams of parts of speech, extracted from POS tagged text. Specifically, we extend the work of [7] who estimate how informative a word sequence corresponding to a POS n-gram can be. We use this to derive a quality score for each term separately, not as a sequence, which we then integrate to the weighting model. Our intuition is that the term frequency statistics used by weighting models could be enhanced to better discriminate between terms, using our proposed measure of term informativeness. Hence, our goal is to assist the lexical features used by these models (e.g. term frequency), which are query/document-dependent, with a non-lexical feature (our proposed term quality), which is query/document-independent.

We evaluate the impact of integrating our proposed term quality into weighting models upon retrieval performance, using the original weighting model as a baseline. Experiments on two established weighting models and two standard TREC collections, show that term quality improves retrieval performance sometimes considerably, and consistently at all times.

The remainder of this paper is organised as follows. Section 2 presents related studies. Section 3 details our methodology for deriving term quality and for extending weighting models with it. Section 4 discusses the evaluation, and Section 5 summarises our conclusions.

2 Related Studies

We propose to extend weighting models with a measure of term quality. The notion of an intrinsic term quality is new in IR (to our knowledge), but not in linguistics. Word commonness, which measures how common a word is generally in language, is used in theoretical and practical linguistics, e.g. in quantitative linguistics, lexicography, and language teaching [16]. Mikk [9] suggests a ‘corrected term frequency’ based on word commonness, which predicts the complexity of a document’s content. Our proposed term quality is based on the same intuition, namely that raw term frequency could be enhanced to better discriminate between terms. Unlike previous studies, we specifically apply this intuition to IR, and ask whether we can extend models that are tailored to process term frequencies, with the proposed term quality, so as to improve retrieval performance.

There are two novel contributions in this work. First, even though previous work has extended weighting models with various types of evidence, as briefly mentioned in Section 1 [4,6,8,15], to our knowledge, no study has reported integrating term evidence that is both query- and document-independent to the weighting model. The closest to this notion is removing stopwords before retrieval, in the sense that words from a standard list are removed regardless of the query/document. The second novelty of our approach is that we use POS n-grams to derive a measure of term quality. POS n-grams were first used in POS tagging, to determine the probability of occurrence of POS tags [3]. More recently, POS n-grams were used to estimate the quality of word sequences in IR [7]. This work is a continuation of the latter, because it uses the quality of POS n-grams to derive a measure of quality for individual terms.

3 Methodology

3.1 Deriving Term Quality from Parts of Speech

We derive a term quality measure from POS n-grams as follows. We use a POS tagger to POS tag a collection of documents. Any POS tagger and any large collection of documents can be used. We extract POS n-grams from each POS-tagged sentence in each document. For example, for a sentence ABCDEF, where parts of speech are denoted by the single letters A, B, C, D, E, F and where POS n-gram length = 4, the POS n-grams extracted are ABCD, BCDE, and CDEF. Then, we compute the quality of each POS n-gram, using the *content load* (*cl*) estimator described in [7]. This estimator considers nouns, verbs, adjectives and participles more informative than other parts of speech, with nouns the most informative, following evidence given in [11,19]. The formula is:

$$cl = \frac{C_N + C_{AVP} \cdot \varrho}{n} \tag{1}$$

where C_N = number of nouns in the POS n-gram, C_{AVP} = number of adjectives and/or verbs and/or participles in the POS n-gram, n = POS n-gram length, and ϱ = penalising variable applied to adjectives and/or verbs and/or participles. The value of ρ is automatically derived from collection statistics [7]. Using Equation (1), the content load of a POS n-gram is between 0 and 1, where 0 and 1 are the minimum and maximum values, respectively. This content load for POS n-grams approximates how important any of the word sequences that correspond to a POS n-gram can be. We extend the IR system's inverted file with information about the POS n-grams that are associated with each term, and their content load. The inverted file of an IR system contains statistics on term frequencies in each document and in the whole collection.

Based on Equation (1), we propose to compute the quality score for each term (*tqs*) as follows:

$$tqs = \frac{\sum cl_t}{f_{POSngram_t}} \tag{2}$$

Table 1. Example term quality for TREC query 451 (stemmed & no stopwords)

term	quality score	term	quality score	term	quality score	term	quality score
bengal	0.46	catteri	0.42	includ	0.19	program	0.29
breed	0.31	characterist	0.31	item	0.33	refer	0.25
breeder	0.42	club	0.35	name	0.25	relev	0.24
carri	0.20	discus	0.35	onli	0.16	tiger	0.39
cat	0.35	discuss	0.19	origin	0.22		

where cl_t is the content load of a POS n-gram that contains[2] term t (computed using Equation (1)), and $f_{POSngram_t}$ is the number of POS n-grams that contain term t. Note that we consider the content load of all POS n-grams, regardless of which documents they occur in, because our goal is to derive a global, as opposed to document-centric, estimation of term quality. Using Equation (2), the term quality score is between 0 and 1, where 0 and 1 are the minimum and maximum scores, respectively. The term quality score approximates how important a term generally is, based on its part of speech and the POS n-grams in which it occurs in the collection.

Table 1 gives an example of the quality score given to terms from TREC query number 451 (queries are presented in Section 4.). Terms are stemmed and stopwords are removed in this example. We see that `bengal`, `breeder` and `catteri` have the highest *tqs*, while `desc`, `includ` and `onli` have the lowest. Even though *tqs* is derived from POS n-grams, and specifically a formula that rewards nouns, slightly penalises adjectives, verbs and participles, and ignores everything else, the term quality score seems to discriminate between terms on the basis of more than just their POS class. Hence, the highest scoring term is an adjective (`bengal`), not a noun, in this query. Similarly, while both `name` and `tiger` are nouns, they have different scores (0.25 and 0.39, respectively). Overall, the main point to remember here is that the quality scores assigned to these query terms have been derived from POS, not lexical, statistics, extracted from a whole collection. Hence, these term quality scores are completely document independent.

POS tagging, extracting POS n-grams and computing term quality take place once at indexing time, with negligible overhead.

3.2 Integrating Term Quality to Term Weighting

Section 3.1 introduced a general quality measure for terms, which is document-independent. More simply, the proposed term quality measures how informative a term generally is, and not how relevant a term is to another. In order for such a general quality measure to be used in retrieval, it needs to be integrated with relevance weighting, i.e. classical term weighting that determines the relevance of a term to a document. We present how we integrate the proposed term quality to term weighting in this section.

[2] POS n-grams contain POS tags, not terms. By term 'contained' in a POS n-gram we mean a term that, when tagged, has its POS tag captured in a POS n-gram.

We estimate the relevance $R(d, Q)$ between a document d and a query Q, as:

$$R(d, Q) = \sum_{t \in Q} w(t, d) \cdot qtw \tag{3}$$

where t is a term in Q, $w(t, d)$ is the weight of term t for a document d, and qtw is the query term weight. $w(t, d)$ can be computed by different weighting models in different ways [1,6,14]. All these models however use the frequency of a term in a document (tf) one way or another. For example, for BM25 [14]:

$$w(t, d) = w^{(1)} \cdot \frac{(k_3 + 1) \cdot qtf}{k_3 + qtf} \cdot tfn \tag{4}$$

where k_3 is a parameter, qtf is the query term frequency, and tfn is the normalised term frequency in a document, given by:

$$tfn = \frac{(k_1 + 1) \cdot tf}{tf + k_1 \cdot (1 - b + b \cdot \frac{l}{avg\_l})} \tag{5}$$

where k_1 & b are parameters, and l (avg_l) is the document length (average document length in the collection).

In Equation (4), $w^{(1)}$ is the weight of a term in the query, given by:

$$w^{(1)} = log \cdot \frac{N - n + 0.5}{n + 0.5} \tag{6}$$

where N is the number of all documents in the collection, and n is the number of documents containing term t. Note that $w^{(1)}$ is the inverse document frequency (*idf*) component of BM25.

To recapitulate, tf is used as an integral part of BM25 to compute the relevance of a document to a query (Equation (5)).

Our aim is to show how $w(t, d)$ can be altered to include our proposed term quality. We extend weighting models with term quality (*tqs*), computed using Equation (2), by altering term frequency (tf) in the $w(t, d)$ component (see Equation (3)) of each weighting model, as follows:

$$tf_q = tf \cdot \frac{1}{1 - tqs} \tag{7}$$

where tf_q is the term frequency that is altered by the term quality score, tf is the original term frequency of term t in a document, and *tqs* is the proposed term quality. The idea here is to boost the discriminative effect of term frequency with knowledge about how informative a term generally is in language. The reason why we use 1 / (1 - *tqs*), instead of raw *tqs*, is explained at the end of this section. The main point to remember here is that by using term quality to alter term frequency, we integrate it into the weighting model as part of the $w(t, d)$ component, and not externally (e.g. as a prior). Note that the integration proposed (Equation (7)) is one among several possible and potentially more

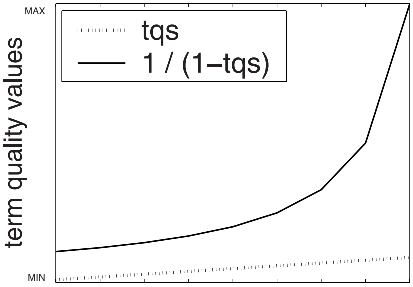

Fig. 1. *tqs* (computed using Equation (2)) across its value range ($min - max$=0-1)

sophisticated ways of integrating term quality to the term weighting model. Our focus is to initially test if term quality works for retrieval, and not yet to optimise its integration into term weighting.

We have shown how we integrate term quality to the weighting model by multiplying it to term frequency before the relevance of a document to a query is 'weighted'. In fact, this integration takes place even before term frequency is normalised with respect to document length. We do not know what the effect of normalisation will be. In this work, we assume that normalising term frequency should not affect the integration of term quality into the model. However, this assumption is worth testing in the future.

Why do we use 1 / (1 - *tqs*) instead of *tqs*? We know that term quality can be between 0 and 1. Figure 1 plots term quality for values within this range. Note that we plot *tqs* as a simple function here, meaning the x-axis is simply the arguments of the function. The distribution of term quality across its value range is the dotted line. We see that a simple transformation of *tqs*, namely 1 / (1 - *tqs*), widens the value range considerably (solid line). We assume that widening the value range of the proposed term quality will render it more discriminative, and this is why we prefer it over the raw *tqs*, when integrating it to the weighting model.

Implementation-wise, integrating term quality to the weighting model takes place when documents are matched to queries, and consists of a simple look-up of POS n-gram statistics in the IR system's inverted file. This is done simultaneously to the usual term statistics look-up, with negligible overhead.

4 Evaluation

We aim to test whether integrating the proposed term quality score into weighting models can enhance retrieval performance. We use two standard TREC collections, namely WT10G (TREC 2000-2001), which contains 1.69 million Web documents (10GB), and Disks 4&5 (TREC 2004), which contain 528 thousand mainly newswire documents (2GB). We remove the Congressional Record from Disks 4&5, according to TREC 2003-2004 settings. We use topics 451-550

Table 2. Weighting model parameters (b for BM25, c for PL2)

WT10G		Disk 4&5	
default	optimal	default	optimal
b=0.75, c=1.00	b=0.27, c=13.13	b=0.75, c=1.00	b=0.34, c=12.00

(WT10G) and topics 301-450 & 601-700 (Disks 4&5) to retrieve relevant documents. We use short topics (title-only) because they are more representative of real user queries. For indexing and retrieval, we use the Terrier IR platform [10], and apply stopword removal and Porter's stemming during indexing.

To compute the term quality score, we POS tag WT10G, using the TreeTagger, because it is fast (∼10,000 tokens/sec) and has a low error rate (∼3.8%) [17]. Following [7], we extract POS n-grams of length n= 4[3] from the collection, and compute their content load using Equation (1), with $\rho = 0.17$.

To match documents to query terms, we use BM25 [14] and PL2 from the Divergence From Randomness (DFR) framework [1]. Each of these models treats the matching process differently, giving us a varied setting for our experiments. Each model has a term frequency normalisation parameter (b for BM25, and c for PL2). These parameters can be tuned according to query/collection statistics, and can affect retrieval performance considerably [1].

To evaluate the impact of integrating term quality to the weighting model upon retrieval performance, with the original weighting models as a baseline, we conduct three series of experiments. Throuhgout, we use the mean average precision (MAP) to evaluate retrieval performance. 1) We set all weighting model parameters to default/recommended values. (See [14] for default b values; c values are recommended at [1] [4].) 2) To test the effect of our approach on a stronger baseline than that of default values, we optimise all weighting model tf normalisation parameters for MAP, by training using data sweeping and simulated annealing over a large range of values. We optimise the baseline for MAP and use the same parameter for the weighting model with the term quality (i.e. we assume that the optimal b value for BM25 will also be optimal for BM25 with term quality). All parameter values used are shown in Table 2. 3) To raise even more the baseline, we use optimal values and query expansion (QE), which is an automatic performance-boosting technique that extracts the most relevant terms from the top retrieved documents, and uses them to expand the initial query. The expanded query is then used to retrieve documents anew. For query expansion, we use the Bo1 DFR weighting model [1], and extract the 30 most relevant terms from the top 5 retrieved documents, which is the recommended setting [1]. Table 3 displays the evaluation results.

Table 3 shows that, at all times, the proposed term quality improves MAP, over the original weighting models, with a statistical significance (Wilcoxon matched-pairs signed-ranks test) for Disks 4&5. This conclusion is consistent

[3] varying n between 3 and 6 gives similar results to the ones reported here.

[4] c values are also recommended at:
http://ir.dcs.gla.ac.uk/terrier/doc/dfr_description.html

Table 3. MAP scores using weighting models with default/optimised parameters and query expansion (QE); baseline = original weighting model; term quality = weighting model with term quality; * (**) = stat. significance at p<0.05 (p<0.01) with Wilcoxon matched-pairs signed-ranks test.

		WT10G		Disks 4&5	
settings	model	baseline	term quality	baseline	term quality
default	BM25	0.1874	0.1923 (+2.6%)	0.2363	0.2425 (+2.6%**)
	PL2	0.1753	0.1846 (+5.3%**)	0.2242	0.2348 (+4.7%**)
optimised	BM25	0.2096	0.2104 (+0.4%)	0.2499	0.2549 (+2.0%**)
	PL2	0.2093	0.2112 (+1.9%)	0.2530	0.2532 (+0.1%*)
optimised + QE	BM25	0.2362	0.2440 (+3.3%)	0.2933	0.2985 (+1.8%**)
	PL2	0.2241	0.2276 (+1.6%)	0.2966	0.2980 (+0.8%**)

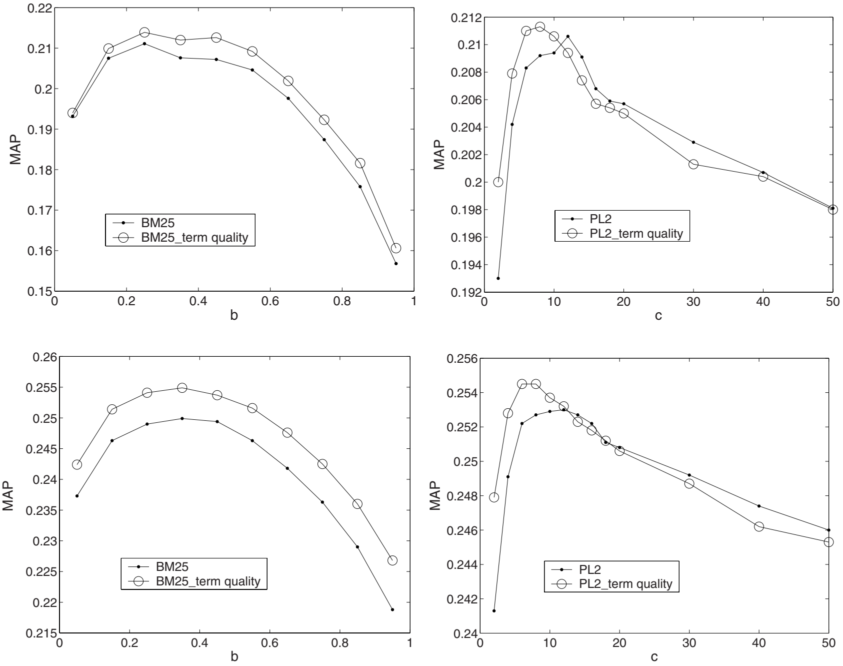

Fig. 2. WT10G in top row, Disks 4&5 in bottom row. The x axis is the weighting model parameter (*b* for BM25, *c* for PL2). The y axis is the Mean Average Precision.

for two different weighting models, with and without query expansion, for 350 topics and for two different collections, hence it is a solid indication that enriching term frequency with the proposed term quality can enhance retrieval. This finding is also supported by Figure 2, which shows the effect of varying the weighting model parameters without term quality (dot) and with term quality (circle), for the two collections. Two trends emerge in Figure 2: 1) the baseline

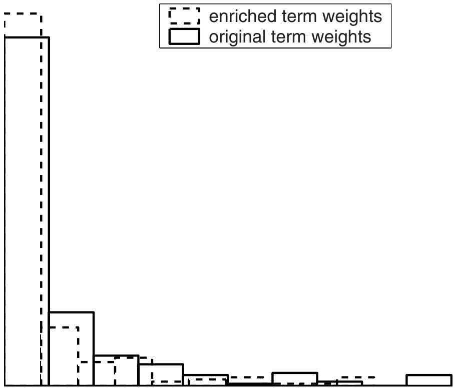

Fig. 3. BM25 term weights for all 350 queries for the top retrieved document, without *tqs* (original) and with *tqs* (enriched)

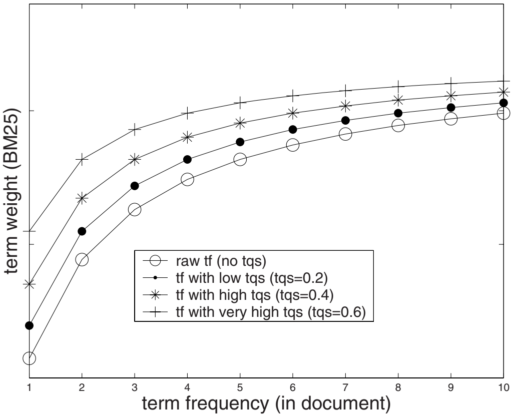

Fig. 4. *tf* component of BM25 term weight, without and with *tqs*

and term quality lines have similar shapes throughout, and 2) the term quality line (circle) moves generally at higher MAP values than the baseline line (dot). This means that our integration of term quality to the weighting model helps retrieval in a consistent way. This consistency can explain the fact that the optimal *tf* normalisation values are very similar and sometimes identical for the baseline and for using term quality.

Figure 3 shows an example of the effect of integrating term quality into BM25 as a histogram. We plot the weight of all 350 queries for the top retrieved document as a function, similarly to Figure 1. We compare the term weights computed by the baseline BM25 (solid line) and BM25 with term quality (dotted line)[5]. High term weights mean that a term is very relevant to the document, and vice

[5] PL2 behaves very similarly.

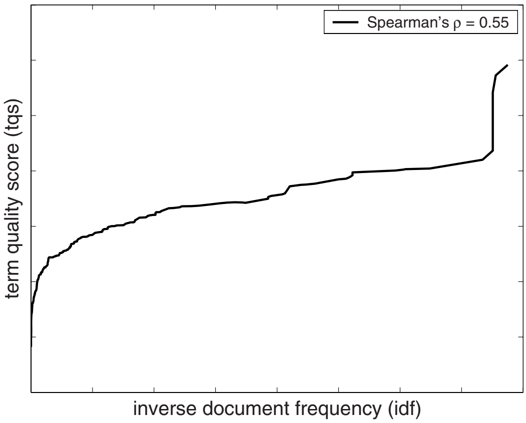

Fig. 5. *idf* versus *tqs* for all query terms used

versa. We see that term quality generally renders high term weights even higher and low term weights even lower. Hence, term quality appears to make the resulting term weights more discriminative, which could explain the improvement in retrieval performance shown in Table 3.

Figure 4 plots the effect of integrating term quality to term frequency, ignoring term frequency normalisation, and for three different term quality values. The x axis is the term frequency, and the y axis is the term weight computed using BM25. We plot the original term frequency (circle), term frequency with low term quality (dot), term frequency with high term quality (star), and term frequency with very high term quality (cross). We see that integrating term quality never breaks the non-linear saturation of term frequency. More simply, the gain (in informativeness) in seeing the term for the first time is much greater, than seeing that term subsequently, even after we have integrated our proposed term quality to term frequency. This is very similar to the effect of inverse document frequency upon term frequency [15] (this point is further discussed in the next paragraph). This shows that term quality is compatible to term frequency. Note that the effect of term quality on term frequency becomes more noticeable as term frequency decreases. More simply, a term with high tf will be boosted less by *tqs*, than a term with low tf. This is expected, given the difference in magnitude between term quality (between 0-1) and term frequency ($\gg$1).

Finally, Figure 5 shows the relation between inverse document frequency (*idf*) and term quality (*tqs*), for the terms of all 350 queries used in these experiments. We see that, indeed, our proposed measure of term quality is correlated to inverse document frequency (Spearman's $\rho = 0.55$), as indicated previously (Figure 4). This correlation indicates that our proposed term quality is compatible to term frequency, and can explain why intergating *tqs* into the weighting model overall enhances retrieval performance.

5 Conclusion

We introduced a novel notion of term quality, which measures how informative a term generally is, regardless of a document/query. We derived this measure using part of speech (POS) information extracted from a corpus as POS n-grams. We tested this term quality in IR, by integrating it to the term frequency component of two weighting models. We reasoned that if this integration resulted in more accurate term weights, and retrieval performance improved, then we could consider term quality as useful evidence. Experimental results on two standard TREC collections, with default and optimal settings, and query expansion, showed that retrieval performance with term quality improved consistently, and with a statistical significance at all times for one collection, over strong baselines.

The main contribution of this work consists in posing the question: Can there be such a thing as an intrinsic notion of term quality? We showed that yes, there can, and also how to practically apply it to enhance retrieval performance. Future work includes exploring the integration of term quality into the retrieval process, as well as evaluating it in other tasks, such as text classification.

References

1. Amati, G.: Probabilistic Models for Information Retrieval based on Divergence from Randomness, PhD Thesis. University of Glasgow, UK (2003)
2. Brin, S., Page, L.: The Anatomy of a Large-Scale Hypertextual Web Search Engine. Computer Networks and ISDN Systems. 30, 107–117 (1998)
3. Brown, P.F., Pietra, V.J.D., deSouza, P.V., Lai, J.C., Mercer, R.L.: Class-Based N-Gram Models of Natural Language. Computational Linguistics 18(4), 467–479 (1992)
4. Craswell, N., Robertson, S., Zaragoza, H., Taylor, M.: Relevance Weighting for Query Independent Evidence. In: 28^{th} Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 416–423. ACM Press, New York (2005)
5. He, B., Ounis, I.: A Study of the Dirichlet Priors for Term Frequency Normalisation. In: He, B., Ounis, I. (eds.) 28^{th} Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 465–471. ACM Press, New York (2005)
6. Kraaij, W., Westerveld, T., Hiemstra, D.: The Importance of Prior Probabilities for Entry Page Search. In: SIGIR. 25^{th} Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 27–34. ACM Press, New York (2002)
7. Lioma, C., Ounis, I.: Light Syntactically-Based Index Pruning for Information Retrieval. In: ECIR 2007. LNCS, vol. 4425, pp. 88–100. Springer, Heidelberg (2007)
8. Macdonald, C., He, B., Plachouras, V., Ounis, I.: University of Glasgow at TREC2005: Experiments in Terabyte and Enterprise tracks with Terrier. In: TREC. 14^{nth} Text REtrieval Conference (2005)
9. Mikk, J.: Prior Knowledge of Text Content and Values of Text Characteristics. Journal of Quantitative Linguistics. 8(1), 67–80 (2001)

10. Ounis, I., Amati, G., Plachouras, V., He, B., Macdonald, C., Lioma, C.: Terrier: A High Performance and Scalable Information Retrieval Platform. In: Association for Computing Machinery (ACM) Conference on Research and Development in Information Retrieval (SIGIR) Workshop on Open Source Information Retrieval (OSIR), pp. 18–24 (2006)
11. Ozmutlu, S., Spink, A., Ozmutlu, H.C.: A Day in the Life of Web Searching: an Exploratory Study. Information Processing & Management 2, 319–345 (2004)
12. Peng, J., Macdonald, C., He, B., Ounis, I.: Combination of Document Priors in Web Information Retrieval. In: RIAO. 8_{th} Large-Scale Semantic Access to Content, pp. 28–39 (2007)
13. Plachouras, V., Ounis, I.: Multinomial Randomness Models for Retrieval with Document Fields. In: ECIR 2007. LNCS, vol. 4425, pp. 88–100. Springer, Heidelberg (2007)
14. Robertson, S., Walker, S.: Some Simple Approximations to the 2-Poisson Model for Probabilistic Weighted Retrieval. In: 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 232–241 (1994)
15. Robertson, S., Zaragoza, H., Taylor, M.: Simple BM25 Extension to Multiple Weighted Fields. In: CIKM. 13^{nth} ACM International Conference on Information and Knowledge Management, pp. 42–49. ACM Press, New York (2004)
16. Savicky, P., Hlavacova, J.: Measures of Word Commonness. Journal of Quantitative Linguistics 9(3), 215–231 (2002)
17. Schmid, H.: Probabilistic Part-of-Speech Tagging Using Decision Trees. In: International Conference on New Methods in Language Processing, pp. 44–49 (1994)
18. van Rijsbergen, C.J.: Information Retrieval. Butterworths, Butterworths, London, UK (1979)
19. Zubov, A.: Formalization of the Procedure of Singling Out of the Basic Text Contents. Journal of Quantitative Linguistics 11(1-2), 33–48 (2004)